CSE- 4029 LAB Assignment - 1&2

Academic year: 2021-2022 **Semester:** WIN

Faculty Name: Prof. BKSP Kumar raju Alluri **Date:** 18/3/2022

Student name: M.Taran **Reg. no.:** 19BCE7346

Exploratory Data analysis

##Step-1 : Took Dataset with both categorical and continuous data values and drew graphs Accordingly.

#link for dataset : https://archive.ics.uci.edu/ml/datasets/wholesale+customers

```
#Importing Data
customers <-
read.csv("D:/users/lenovo/OneDrive/Desktop/19BCE7346/customer_data.csv",
stringsAsFactors = FALSE)
head(customers, n = 6)

glimpse(customers)
dim(customers)

#Understanding data
ggplot(customers, aes(y = Income)) + geom_boxplot()

# Boxplot of the Year of birth variable
ggplot(customers, aes(Year_Birth)) + geom_boxplot()
```

#Step-2: Apply any ML- Algorithm without pre-processing

#Step-3: Perform Various Categorical and Continuous Encodings

#Encoding the categorical features & continuous

```
# Encoding the categorical features to numeric
customers_copy <- customers_copy %>% mutate(Education =
case_when(Education == "graduate" ~ 1,
                                          Education == "non-graduate" ~ 0))
customers_copy <- customers_copy %>% mutate(Marital_Status =
case_when(Marital_Status == "Taken" ~ 1,
                                             Marital_Status == "Single" ~ 0))

str(customers_copy$Education)
str(customers_copy$Marital_Status)

library(caret)
#preprocessing the data
customers_copy_pre <- preProcess(customers_copy[,c(3, 6:17, 25:26)], method =
c("center", "scale"))

#normalizing
customers_copy <- predict(customers_copy_pre, customers_copy[, c(3, 6:17,
25:26)])
summary(customers_copy)
```

##Step-4: Feature Generation

```
# Going to drop the rows that have missing income values.
customers <- na.omit(customers)
dim(customers)

library(cluster)
sil_width <- map_dbl(2:10, function(k){
  model <- pam(customers_copy, k = k)
  model$silinfo$avg.width
})
sil_df <- data.frame(
  k = 2:10,
  sil_width = sil_width
)
 head(sil_df)
```

```
ggplot(sil_df, aes(k, sil_width)) + geom_line() + scale_x_continuous(breaks = 2:10) +
labs(y = "Avg sil width")
trainingSet_scaled <- as.data.frame(scale(trainingSet[,
getIndipendentNumbersOfCol()]))
testSet_scaled <- as.data.frame(scale(testSet[, getIndipendentNumbersOfCol()]))
dataSet_scaled <- as.data.frame(scale(dataSet[, getIndipendentNumbersOfCol()]))
```

#Step-5: Dimensionality Reduction

```
#Using popular method of dimensionality reduction is the principal component
analysis(PCA)
library(FactoMineR)

#Running a PCA.
customers_copy_pca <- PCA(customers_copy, graph = FALSE)

#Exploring PCA()

# Getting the summary of the pca
summary(customers_copy_pca)

#Getting the variance of the first 7 new dimensions
customers_copy_pca$eig[,2][1:7]

#Getting the cummulative variance
customers_copy_pca$eig[,3][1:7]

#Getting the most correlated variables
dimdesc(customers_copy_pca, axes = 1:2)


#Tracing variable contributions in customers_pca
customers_copy_pca$var$contrib

#Visualising PCA
library(factoextra)
#Barplotting the contributions of variables
```

```
fviz_contrib(customers_copy_pca, choice = "var", axes = 1, top = 5)

#Biplots

fviz_pca_biplot(customers_copy_pca)
```

#Step-6: Creating Missing values and identifying the best missing value technique

```
#counting the total number of missing values in the data
library(naniar)
n_miss(customers)

# Summarizing missingness in each variable
miss_var_summary(customers)

#Pre-Processing Data
# Going to drop the rows that have missing income values.
customers <- na.omit(customers)
dim(customers)

#Parsing the Dt_Customer as Date object
#But first i need to make sure that the date is according to ISO 8601 standards
before converting it to a Date object thats where the function dmy() from the
lubridate package comes in.
library(lubridate)
customers <- customers %>% mutate(Dt_Customer = as.Date(dmy(Dt_Customer)))
str(customers$Dt_Customer)
# Dates of the oldest and newest recorded customer
paste0("The oldest enrolment date of a customer dates to: ",
min(customers$Dt_Customer))
paste0("The newest enrolment date of a customer dates to: ",
max(customers$Dt_Customer))


#creating a new variable Age from Year of Birth
customers <- customers %>% mutate(Age = 2021 - Year_Birth)
```

```
customers %>% select(Age) %>% arrange(desc(Age)) %>% top_n(3)
# Max Age is > 100
#Dropping outliers by setting a cap on Income and Age
customers <- customers %>% filter(Income < 600000 & Age < 90)
dim(customers)
```

#Step-7: Applying k-means algorithm on the final dataset and comparing the performance in step-2 and current performance.

```
library(corrplot)

#Getting correlation matrix
cust_cor <- cor(customers[,3:17])
corrplot(cust_cor, method = "color", addCoef.col = "white")

#The elbow method

library(purrr)
tot_withinss <- map_dbl(1:10, function(k){
  model <- kmeans(x = customers_copy, centers = k)
  model$tot.withinss
})

elbow_df <- data.frame(
      k = 1:10,
      tot_withinss = tot_withinss)
head(elbow_df)
 #plotting the elbow plot
ggplot(elbow_df, aes(k, tot_withinss)) + geom_line() + scale_x_continuous(breaks =
1:10)

#Visualizing the top 5 features in the contribution

#visualizing wines
customers %>% ggplot(aes(wines)) + geom_histogram(color = "black", fill =
"lightblue") + facet_wrap(vars(cluster))
#visualizing Income variable
```

```
customers %>% ggplot(aes(Income)) + geom_histogram(color = "black", fill =
"lightgreen") + facet_wrap(vars(cluster)) +
geom_vline(aes(xintercept=mean(Income)),color="blue", linetype="dashed", size =
1)
#visualizing Total_spent
customers %>% ggplot(aes(Total_spent)) + geom_histogram(color = "black", fill =
"purple") + facet_wrap(vars(cluster))
#visualizing NumCatalogPurchases
customers %>% ggplot(aes(NumCatalogPurchases)) + geom_histogram(color =
"black", fill = "orange") + facet_wrap(vars(cluster))
#visualizing meat variable
customers %>% ggplot(aes(meat)) + geom_histogram(color = "black", fill =
"brown") + facet_wrap(vars(cluster))
```